CLAIMS

What is claimed is:

1. A data dependent scrambler (DDS) for a communications channel that transmits a user data sequence having a plurality of symbols, comprising:

a scrambler that generates a scrambled user data sequence that is based on said user data sequence and a seed; and

a first encoder that selectively interleaves adjacent symbols in said scrambled user data sequence if an all-zero symbol is produced by bit interleaving, that identifies a pivot bit that is adjacent to said all-zero symbol if interleaving is performed, and that replaces said all-zero symbol with an all-one symbol if said pivot bit is zero.

2. The DDS of Claim 1 further comprising a seed finder that selects said seed based on said symbols of said user data sequence.

3. The DDS of Claim 1 wherein said scrambler performs a bit-wise XOR operation on said user data sequence and said seed to generate said scrambled user data sequence.

4. The DDS of Claim 1 further comprising a code finder that selects an H-code based on symbols of said user data sequence.

5. The DDS of Claim 4 further comprising a second encoder that reduces a Hamming weight of said scrambled user data sequence using said H-code, said seed and said scrambled user data sequence.

6. The DDS of Claim 5 wherein said second encoder generates a first token and a second token.

7. The DDS of Claim 6 wherein said first token is determined based on said seed and said H-code.

8. The DDS of Claim 7 wherein said first token is determined by a bit-wise XOR operation on said seed and said H-code.

9. The DDS of Claim 6 wherein said second token is a ones complement of said first token.

10. The DDS of Claim 5 wherein said second encoder includes a symbol lookup table, processes two adjacent symbols of said scrambled user data sequence at a time and selectively employs said first and second tokens and said symbol lookup table to encode said scrambled user data based on a Hamming weight of said two adjacent symbols.

11. The DDS of Claim 1 wherein said DDS is implemented in a write path of a data storage device.

12. A communications channel that transmits a user data sequence having a plurality of symbols, comprising:

a host bus interface (HBI) that receives said user data sequence; and

a data dependent scrambler (DDS) that receives said user data sequence from said HBI and that includes:

a scrambler that generates a scrambled user data sequence that is based on said user data sequence and a seed; and

a first encoder that selectively interleaves adjacent symbols in said scrambled user data sequence if an all-zero symbol is produced by bit interleaving, that identifies a pivot bit that is adjacent to said all-zero symbol if interleaving is performed, and that replaces said all-zero symbol with an all-one symbol if said pivot bit is zero.

13. The communications channel of Claim 12 further comprising an error correction coding and cyclic redundancy check (ECC/CRC) device that generates ECC and CRC bits that are based on said scrambled user data sequence.

14. The communications channel of Claim 13 further comprising a run length limited (RLL) coding device that generates an RLL sequence based on said ECC bits and said CRC bits.

15. The communications channel of Claim 12 wherein said DDS further comprises a seed finder that selects said seed based on said symbols of said user data sequence.

16. The communications channel Claim 12 wherein said scrambler performs a bit-wise XOR operation on said user data sequence and said seed to generate said scrambled user data sequence.

17. The communications channel of Claim 12 wherein said DDS further comprises a code finder that selects an H-code based on symbols of said user data sequence.

18. The communications channel of Claim 17 wherein said DDS further comprises a second encoder that reduces a Hamming weight of said scrambled user data sequence using said H-code, said seed and said scrambled user data sequence.

19. The communications channel Claim 18 wherein said second encoder generates a first token and a second token.

20. The communications channel Claim 19 wherein said first token is determined based on said seed and said H-code.

21. The communications channel Claim 20 wherein said first token is determined by a bit-wise XOR operation on said seed and said H-code.

22. The communications channel Claim 19 wherein said second token is a ones complement of said first token.

23. The communications channel Claim 18 wherein said second encoder includes a symbol lookup table, processes two adjacent symbols of said scrambled user data sequence at a time and selectively employs said first and second tokens and said symbol lookup table to encode said scrambled user data based on a Hamming weight of said two adjacent symbols.

24. The communications channel Claim 12 wherein said DDS is implemented in a write path of a data storage device.

25. A method of scrambling a user data sequence that has a plurality of symbols and that is transmitted through a communications channel, comprising:

scrambling said user data sequence based on said user data sequence and a seed to provide a scrambled user data sequence;

interleaving adjacent symbols in said scrambled user data sequence if an all-zero symbol is produced by bit interleaving;

identifying a pivot bit that is adjacent to said all-zero symbol if interleaving is performed; and

replacing said all-zero symbol with an all-one symbol if said pivot bit is zero.

26. The method of Claim 25 further comprising selecting said seed based on said symbols of said user data sequence.

27. The method of Claim 25 wherein said scrambling is performed using a bit-wise XOR operation on said user data sequence and said seed.

28. The method of Claim 25 further comprising selecting an H-code based on symbols of said user data sequence.

29. The method of claim 28 further comprising reducing a Hamming weight of said scrambled user data sequence using said H-code, said seed and said scrambled user data sequence.

30. The method of Claim 29 wherein said reducing step is performed based on a first token and a second token.

31. The method of Claim 30 wherein said first token is determined based on said seed and said H-code.

32. The method of Claim 31 further comprising determining said first token by a bit-wise XOR operation on said seed and said H-code.

33. The method of Claim 30 wherein said second token is a ones complement of said first token.

34. The method of Claim 29 further comprising:

processing two adjacent symbols of said scrambled user data sequence at a time; and

employing said first and second tokens and a symbol lookup table to selectively encode said scrambled user data based on a Hamming weight of said two adjacent symbols.

35. A method of transmitting a user data sequence having a plurality of symbols over a communications channel, comprising:

receiving said user data sequence at a host bus interface (HBI);

relaying said user data sequence from said HBI to a data dependent scrambler (DDS);

scrambling said user data sequence based on said user data sequence and a seed to provide a scrambled user data sequence; and

interleaving adjacent symbols in said scrambled user data sequence if an all-zero symbol is produced by bit interleaving;

identifying a pivot bit that is adjacent to said all-zero symbol if interleaving is performed; and

replacing said all-zero symbol with an all-one symbol if said pivot bit is zero.

36. The method of Claim 35 further comprising generating error correction coding (ECC) and cyclic redundancy check (CRC) bits based on said scrambled user data sequence.

37. The method of Claim 36 further comprising generating a run length limited (RLL) sequence based on said ECC bits and said CRC bits.

38. The method of Claim 35 further comprising selecting said seed based on said symbols of said user data sequence.

39. The method of Claim 35 wherein said scrambling is performed using a bit-wise XOR operation on said user data sequence and said seed.

40. The method of Claim 35 further comprising selecting an H-code based on symbols of said user data sequence.

41. The method of claim 40 further comprising reducing a Hamming weight of said scrambled user data sequence using said H-code, said seed and said scrambled user data sequence.

42. The method of Claim 41 wherein said reducing step is based on a first token and a second token.

43. The method of Claim 42 wherein said first token is determined based on said seed and said H-code.

44. The method of Claim 43 further comprising determining said first token by a bit-wise XOR operation on said seed and said H-code.

45. The method of Claim 42 wherein said second token is a ones complement of said first token.

46. The method of Claim 41 further comprising:

providing a symbol lookup table;

processing two adjacent symbols of said scrambled user data sequence at a time; and

employing said first and second tokens and said symbol lookup table to selectively encode said scrambled user data based on a Hamming weight of said two adjacent symbols.

47. A data dependent scrambler (DDS) for a communications channel that transmits a user data sequence having a plurality of symbols, comprising:

scrambling means for generating a scrambled user data sequence that is based on said user data sequence and a seed; and

first encoding means for selectively interleaving adjacent symbols in said scrambled user data sequence if an all-zero symbol is produced by bit interleaving, for identifying a pivot bit that is adjacent to said all-zero symbol if interleaving is performed, and for replacing said all-zero symbol with an all-one symbol if said pivot bit is zero.

48. The DDS of Claim 47 further comprising seed finding means for selecting said seed based on said symbols of said user data sequence.

49. The DDS of Claim 47 wherein said scrambling means performs a bit-wise XOR operation on said user data sequence and said seed to generate said scrambled user data sequence.

50. The DDS of Claim 47 further comprising code finding means for selecting an H-code based on symbols of said user data sequence.

51. The DDS of Claim 50 further comprising second encoding means for reducing a Hamming weight of said scrambled user data sequence using said H-code, said seed and said scrambled user data sequence.

52. The DDS of Claim 51 wherein said second encoding means generates a first token and a second token.

53. The DDS of Claim 52 wherein said first token is determined based on said seed and said H-code.

54. The DDS of Claim 53 wherein said first token is determined by a bit-wise XOR operation on said seed and said H-code.

55. The DDS of Claim 52 wherein said second token is a ones complement of said first token.

56. The DDS of Claim 51 wherein said second encoding means includes a symbol lookup table, processes two adjacent symbols of said scrambled user data sequence at a time and selectively employs said first and second tokens and said symbol lookup table to encode said scrambled user data based on a Hamming weight of said two adjacent symbols.

57. The DDS of Claim 47 wherein said DDS is implemented in a write path of a data storage device.

58. A communications channel that transmits a user data sequence having a plurality of symbols, comprising:

interface means for receiving said user data sequence; and

data dependent scrambling means for receiving said user data sequence from said interface means and including:

scrambling means for producing a scrambled user data sequence that is based on said user data sequence and a seed; and

first encoding means for selectively interleaving adjacent symbols in said scrambled user data sequence if an all-zero symbol is produced by bit interleaving, for identifying a pivot bit that is adjacent to said all-zero symbol if interleaving is performed, and for replacing said all-zero symbol with an all-one symbol if said pivot bit is zero.

59. The communications channel of Claim 58 further comprising error correction coding and cyclic redundancy check (ECC/CRC) means for generating ECC and CRC bits that are based on said scrambled user data sequence.

60. The communications channel of Claim 59 further comprising run length limited (RLL) coding means for generating an RLL sequence based on said ECC bits and said CRC bits.

61. The communications channel of Claim 58 wherein said data dependent scrambling means further comprises seed finding means for selecting said seed based on said symbols of said user data sequence.

62. The communications channel Claim 58 wherein said scrambling means performs a bit-wise XOR operation on said user data sequence and said seed to generate said scrambled user data sequence.

63. The communications channel of Claim 58 wherein said data dependent scrambling means further comprises code finding means for selecting an H-code based on symbols of said user data sequence.

64. The communications channel of Claim 63 wherein said data dependent scrambling means further comprises second encoding means for reducing a Hamming weight of said scrambled user data sequence using said H-code, said seed and said scrambled user data sequence.

65. The communications channel Claim 64 wherein said second encoding means generates a first token and a second token.

66. The communications channel Claim 65 wherein said first token is determined based on said seed and said H-code.

67. The communications channel Claim 66 wherein said first token is determined by a bit-wise XOR operation on said seed and said H-code.

68. The communications channel Claim 65 wherein said second token is a ones complement of said first token.

69. The communications channel Claim 64 wherein said second encoding means includes a symbol lookup table, processes two adjacent symbols of said scrambled user data sequence at a time and selectively employs said first and second tokens and said symbol lookup table to encode said scrambled user data based on a Hamming weight of said two adjacent symbols.

70. The communications channel Claim 58 wherein said data dependent scrambling means is implemented in a write path of a data storage device.

71. A data dependent descrambler (DDD) for a communications channel that receives a scrambled user data sequence having a plurality of symbols and a seed, comprising:

a first decoder that processes pairs of said symbols, that selectively interleaves adjacent symbols in said scrambled user data sequence when an all-zero symbol is present and that replaces an all-one symbol with an all-zero symbol and interleaves said adjacent symbols when one of said adjacent symbols is an all-one symbol and another of said adjacent symbols is not an all-zero symbol; and

a descrambler that generates a user data sequence that is based on said scrambled user data sequence and said seed.

72. The DDD of Claim 71 wherein said descrambler performs a bit-wise XOR operation on said scrambled user data sequence and said seed to generate said user data sequence.

73. The DDD of Claim 71 wherein said DDD receives an H-code and further comprising a second decoder that selectively decreases a Hamming weight of said scrambled user data sequence using said H-code, said seed and said scrambled user data sequence.

74. The DDD of Claim 73 wherein said second decoder generates a first token and a second token.

75. The DDD of Claim 74 wherein said first token is determined based on said seed and said H-code.

76. The DDD of Claim 75 wherein said first token is determined by a bit-wise XOR operation on said seed and said H-code.

77. The DDD of Claim 74 wherein said second token is a ones complement of said first token.

78. The DDD of Claim 73 wherein said second decoder includes a symbol lookup table, processes two adjacent symbols of said scrambled user data sequence at a time and selectively employs first and second tokens and said symbol lookup table to decode said scrambled user data.

79. The DDD of Claim 71 wherein said DDS is implemented in a read path of a data storage device.

80. A method of descrambling a scrambled user data sequence that has a plurality of symbols and a seed and that is received by a communications channel, comprising:

processing pairs of said symbols;

interleaving adjacent symbols in said scrambled user data sequence when an all-zero symbol is present; and

replacing an all-one symbol with an all-zero symbol and interleaving said adjacent symbols when one of said adjacent symbols is an all-one symbol and another of said adjacent symbols is not an all-zero symbol.

81. The method of Claim 80 further comprising descrambling said scrambled user data sequence based on said user data sequence and a seed to provide a user data sequence.

82. The method of Claim 81 wherein said descrambling step is performed using a bit-wise XOR operation on said scrambled user data sequence and said seed.

83. The method of Claim 80 wherein said communications channel also receives an H-code.

84. The method of claim 83 further comprising selectively decreasing a Hamming weight of said scrambled user data sequence using said H-code, said seed and said scrambled user data sequence.

85. The method of Claim 84 wherein said decreasing step is performed based on a first token and a second token.

86. The method of Claim 85 wherein said first token is determined based on said seed and said H-code.

87. The method of Claim 86 further comprising determining said first token by a bit-wise XOR operation on said seed and said H-code.

88. The method of Claim 85 wherein said second token is a ones complement of said first token.

89. The method of Claim 84 further comprising:

processing two adjacent symbols of said scrambled user data sequence at a time; and

employing said first and second tokens and a symbol lookup table to selectively decode said scrambled user data.

90. A data dependent descrambler (DDD) for a communications channel that receives a scrambled user data sequence having a plurality of symbols and a seed, comprising:

first decoding means that processes pairs of said symbols, that selectively interleaves adjacent symbols in said scrambled user data sequence when an all-zero symbol is present and that replaces an all-one symbol with an all-zero symbol and interleaves said adjacent symbols when one of said adjacent symbols is an all-one symbol and another of said adjacent symbols is not an all-zero symbol; and

descrambling means that generates a user data sequence that is based on said scrambled user data sequence and said seed.

91. The DDD of Claim 90 wherein said descrambling means performs a bit-wise XOR operation on said scrambled user data sequence and said seed to generate said user data sequence.

92. The DDD of Claim 90 wherein said communications channel receives an H-code.

93. The DDD of Claim 92 further comprising second encoding means for selectively decreasing a Hamming weight of said scrambled user data sequence using said H-code, said seed and said scrambled user data sequence.

94. The DDD of Claim 93 wherein said second decoding means generates a first token and a second token.

95. The DDD of Claim 94 wherein said first token is determined based on said seed and said H-code.

96. The DDD of Claim 95 wherein said first token is determined by a bit-wise XOR operation on said seed and said H-code.

97. The DDD of Claim 94 wherein said second token is a ones complement of said first token.

98. The DDD of Claim 93 wherein said second decoding means includes a symbol lookup table, processes two adjacent symbols of said scrambled user data sequence at a time and selectively employs said first and second tokens and said symbol lookup table to encode said scrambled user data based on a Hamming weight of said two adjacent symbols.

99. The DDD of Claim 90 wherein said DDS is implemented in a read path of a data storage device.